
```

SCAL RLIN
SCAL 79H
RST 18H:DEFB 79H
DF 79

```

RLIN examines a line on the video display and extracts the arguments from the line.

On entry, register pair DE points to the memory location which is the start of the line.

On exit, The first argument is held in workspace location ARG1 (OCOC/D), the second argument in ARG2 (OCOE/F), etc. The number of arguments is held in ARGN (OCOB).

If more than ten arguments or a non-hexadecimal argument is encountered, the Carry flag is set.

ie.

X	aaaa	bbbb	cccc	jjjj	n arguments
(DE)	ARG1	ARG2	ARG3		ARG10	ARGN

```

RST ROUT
RST 30H
F7

```

ROUT writes the character whose ASCII code is in register A to the output devices in the output tble.

The character may be a screen edit command, eg 11H (CUL, cursor left), 16H (CSR, move rest of line to the right so creating a space).

```

SCAL xx
RST 18H:DEFB xx
DF xx

```

SCAL (Subroutine CALL) is the routine by which all the other routines (with the exception of the RSTs) are called. The number of the called routine is xx. Thus, SCAL RLIN = SCAL 79H calls the routine RLIN.

The byte following the SCAL (DFH) is a displacement down a table which contains the actual addresses of the NAS-SYS routines. By using this technique other versions of NAS-SYS may be produced while still retaining compatability with the previous versions, ie. programs using a certain routine in one version will still function correctly when using the same routine in another version.

NAS-SYS 3 only

```

SCAL SCALI
SCAL 7FH
RST 18H:DEFB 7FH
DF 7F

```

SCALI is almost the same as SCALJ. The only difference is that

the number of the routine to be called must be in register E.

```
SCAL SCALJ
SCAL 5CH
RST 18H:DEFB 5CH
DF 5C
```

SCALJ calls the routine whose number is in workspace location ARGC (OCD0A).

eg. If ARGC contains 54H, the Tabulate command is called since the Tabulate routine is numbered 54H, (the ASCII code for T).

The arguments for the command must be stored in workspace locations ARG1, ARG2, etc and the first three arguments must be in HL, DE, and BC.

See also SCALI.

```
SCAL SOUT
SCAL 6DH
RST 18H:DEFB 6DH
DF 6D
```

SOUT transmits a series of bytes to the serial output port.

The start address of the string must be loaded into register pair HL and the length of the string in register B. At the end of transmission, register C holds the eight bit checksum of all the bytes transmitted.

NAS-SYS 3 only

```
SCAL SP2
SCAL 7EH
RST 18H:DEFB 7EH
DF 7E
```

SP2 is the equivalent of two SPACES, described in the following paragraph.

```
SCAL SPACE
SCAL 69H
DF 69
```

SPACE transmits the byte 20H (ASCII code for a space) to all the devices in the output table.

This routine simply loads register A with 20H and then calls ROUT.

See also SP2.

SCAL SRLIN
SCAL 70H
RST 18H:DEFB 70H
DF 70

SRLIN scans the serial input port once. If data is ready, the routine returns with the data in register A and the Carry flag set. Otherwise, the Carry flag is cleared.

The serial port may be connected to a variety of different devices. Refer to Chapter 1 for the physical connections and speed switch settings.

SCAL SRLX
SCAL 6FH
RST 18H:DEFB 6FH
DF 6F

SRLX transfers the contents of register A to the serial output port and waits for transmission to be completed.
See Chapter 1 for setting up the serial port.

SCAL TBCD2
SCAL 67H
RST 18H:DEFB 67H
DF 67

TBCD2 transmits the ASCII codes for the hexadecimal representation of the contents of register A to the devices in the output table. The byte in register A is added into register C, so accumulating an eight bit checksum.

The video display shows the contents of register A as two hexadecimal digits. The serial output port transmits the ASCII codes for the two digits.

SCAL TBCD3
SCAL 66H
RST 18H:DEFB 66H
DF 66

TBCD3 transmits the ASCII codes for the hexadecimal representation of the contents of register pair HL to the devices in the output table. These bytes are followed by the ASCII code for a space (20H). The contents of registers H and L are added into register C which thus accumulates an eight bit checksum.

SCAL TDEL
SCAL 5DH
RST 18H:DEFB 5DH
DF 5D

TDEL generates a delay of 2.9s if the Z80 clock rate is 2MHz. At 4MHz the delay is halved.

```
SCAL TX1
SCAL 6CH
RST 18H:DEFB 6CH
DF 6C
```

TX1 transmits (to the devices in the output table) the ASCII codes for the hexadecimal representation of the contents of register pair HL, followed by the ASCII code for a space (20H), followed by the ASCII codes for the hexadecimal representation of the contents of register pair DE, followed by the ASCII code for a space.

The contents of registers H,L,D, and E are added into register C which thus accumulates an eight bit checksum.

```
SCAL UIN
SCAL 76H
RST 18H:DEFB 76H
DF 76
```

UIN allows a user-written input device driver to be added to the input table.

UIN causes a JUMP to workspace location \$UIN (0C7A). This location contains C3H, the op code for the Z80 JP instruction. The two following bytes are initialised by NAS-SYS to an address within the ROM which contains the byte C9H, the op code for the Z80 RET instruction.

A user-written input device handler may be called by placing the handler start address in the two bytes following \$UIN, ie.0C7B/C. Exit from the handler should be via one of the RET instructions with the Carry flag set only if an input was received.

See UOUT.

```
SCAL UOUT
SCAL 75H
RST 18H:DEFB 75H
DF 75
```

UOUT allows a user-written output device handler to be accessed via the output table. The start address of the handler should be loaded into workspace locations 0C78/9. These locations are initialised by NAS-SYS to the address of a location within the ROM containing the op code for the Z80 RET instruction. Thus UOUT has no effect unless the address is changed.

See UIN.

```
SCAL XOUT
SCAL 6EH
RST 18H:DEFB 6EH
DF 6E
```

XOUT is the handler for output to an external serial device. Individual bits of workspace location \$XOPT (0C28) determine the characteristics of the handler, as follows:-

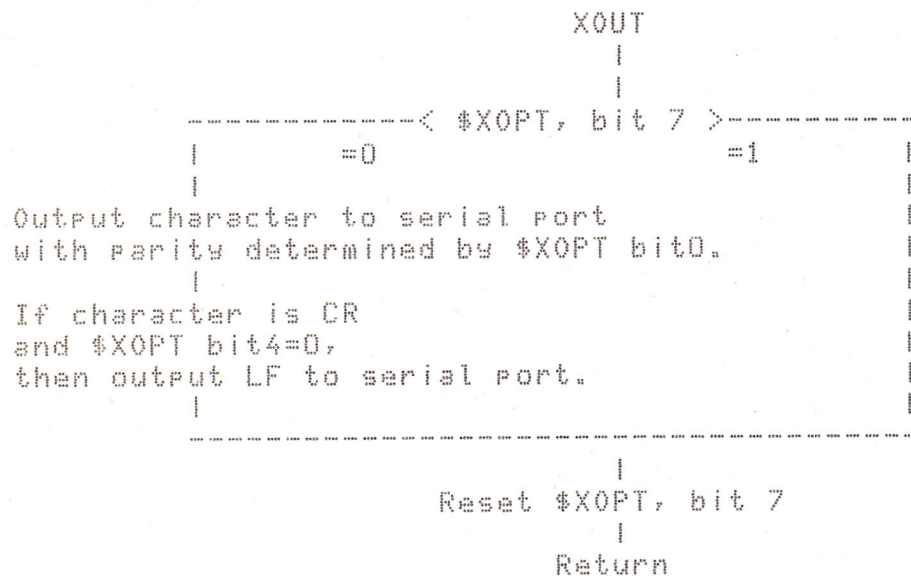
A Guide to NAS-SYS

Bit 0=0 transmit even parity.
1 transmit odd parity.

Bit 4=0 if CR is transmitted, also transmit LF.
1 no LF appended after CR.

Bit 7=0 output characters are transmitted.
1 output characters are not transmitted.

Flowchart:



See also the X command, chapter 2.